Técnicas y Herramientas para Regular la Seguridad en Web Services Basados en WSDL

Hernán Bernardis[(1)], Edgardo Bernardis[(1)], Mario M. Berón[(1)], Daniel E. Riesco[(1)], Maria Joao V. Pereira[(2)]

[(1)]Departamento de Informática / Facultad Ciencias Físico Matemáticas y Naturales/ Universidad Nacional de San Luis
Ejército de los Andes 950 – San Luis – Argentina
{hbernardis, ebernardis, mberon, driesco}@unsl.edu.ar
[(2)] Departamento de Informática e Comunicações/ Instituto Politécnico de Bragança
Bragança - Portugal
mjoao@ipb.pt

Resumen

El desarrollo de sistemas en la actualidad ha mutado, siguiendo la tendencia mundial de migración hacia la nube. Se logra mayor escalabilidad al construir sistemas con módulos distribuidos en la red, en donde el sistema final es la combinación de un conjunto de módulos distribuidos en la nube. Este concepto ha funcionado a lo largo de la historia por medio de diferentes arquitecturas tecnológicas - RPC, RMI, etc. - pero, desde hace algunos años, los servicios web han sido la arquitectura de mayor popularidad. Su independencia de la arquitectura tecnológica subyacente junto con su aprovechamiento de los protocolos ya definidos de internet son algunos de los motivos impulsores de su popularidad.

En el caso particular de esta investigación, el objeto de estudio son los servicios web con descripciones en WSDL. La idea de este trabajo consiste en detectar y - de ser necesario - modificar el nivel de entendimiento que posee la descripción de un servicio web para minimizar las vulnerabilidades que este pueda tener al estar publicado en la nube. Para detectar estas vulnerabilidades, es primordial primero comprender el servicio web y, por ende, su descripción.

En este artículo se describe una línea de investigación centrada en facilitar la comprensión de Servicios Web mediante el análisis de sus especificaciones WSDL. Este análisis sirve como base para el cálculo del grado de entendimiento del mismo y, en base a esta medida, determinar qué acciones se deben tomar, en caso de que corresponda, para cambiar su grado de entendimiento ocultando vulnerabilidades.

Palabras clave: Web Services, Métricas, WSDL, **C**omprensión, Seguridad.

Contexto

La línea de investigación descripta en este artículo se desarrolla en el **La**boratorio de **C**alidad e **I**ngeniería de **S**oftware (**LaCIS**) de la Universidad Nacional de San Luis (UNSL); y se encuentra enmarcada dentro de dos proyectos. "*Ingeniería de Software: Conceptos, Prácticas y Herramientas para el desarrollo de software de Calidad*", perteneciente a la UNSL. Proyecto reconocido por el programa de incentivos y continuación de diferentes proyectos de investigación de gran éxito a nivel nacional e internacional. También dentro del proyecto bilateral con la Universidade do Minho (Portugal): "*Fortalecimiento de la Seguridad de los Sistemas de Software*

mediante el uso de Métodos, Técnicas y Herramientas de Ingeniería Reversa", recientemente aprobado por el Ministerio de Ciencia Tecnología e Innovación Productiva (Mincyt) y su código es PO/16/93.

Introducción

Actualmente, con el auge de internet se están popularizando los Web Services como artefactos de software a partir de los cuales se pueden construir sistemas más complejos. Haciendo una burda comparación, los web services son los ladrillitos Lego® de la programación, que al unirse permiten crear sistemas mayores y muy potentes.

Según la W3C, un Web Service es: "*Una aplicación de software identificada por una URI, cuya interface y enlaces son capaces de ser definidos, descriptos y descubiertos como artefactos XML. Un web service soporta interacción directa con otros agentes de software usando mensajes basados en XML intercambiados a través de protocolos basados en internet"*. Muchas organizaciones construyen sus sistemas basándose en una arquitectura orientada a servicios web, en donde algunos de ellos se publican al resto del mundo y otros simplemente son utilizados de manera interna por sus equipos de desarrollo. Esta descentralización permite que cada equipo de desarrollo elija la arquitectura que desee para construir sus proyectos sin afectar la vinculación con el resto del sistema. La interacción entre proyectos se convierte en un intercambio de mensajes con la información necesaria dentro, sin necesidad de vinculación a nivel de arquitectura subyacente más que la necesaria a la invocación de los servicios web. Es justamente esta vinculación la que planteó el desafío más complejo en el mundo de los servicios web: de qué manera lograr quitar la arquitectura tecnológica subyacente de los sistemas para que la interacción no se viera afectada?. La creación de protocolos específicos de descripción junto al uso de protocolos de internet fueron la solución.

La idea de construir un Web Service y que pueda ser utilizado por cualquier otra persona u organización en el mundo ha sido posible debido a la creación de estándares y lenguajes formales de definición de los mismos. Sin embargo, esta alta abstracción en la construcción y especificación de los mismos dificulta en gran medida su comprensión. Comprensión necesaria tanto a la hora de realizar tareas de mantenimiento (adaptación, arreglo de errores, migración, etc.) como a la hora de analizar las vulnerabilidades que estos servicios pueden dejar al descubierto hacia el mundo. Es importante recordar que un servicio web es una "interfaz" al mundo de un sistema de software que puede ser atacado y vulnerado.

Todo Web Service posee una especificación que provee la información necesaria para invocarlo. Uno de los estándares de descripción más conocido es WSDL (Web Service Definition Language) [1]. Las especificaciones WSDL son un dialecto XML, con reglas bien definidas para especificar cada componente del WS. Cuántos parámetros recibe y de qué tipo son, qué datos retorna y de qué tipo, qué protocolo de internet usa para su comunicación, qué operaciones posee, son entre otras tantas, características del WS que se encuentran especificadas en su WSDL asociado.

Así como el archivo WSDL sirve para que un agente de software o persona pueda interpretarlo para usar el servicio web que describe, también puede dar información a personas no deseadas o incluso exponer vulnerabilidades. Más aún si se considera que existen herramientas que generan los WSDLs de manera automática para un servicio web, con lo cual el nivel de atención a la información que se publica no siempre se encuentra bajo un estricto control. Esto se vuelve más importante

para aquellos casos en donde los servicios web pertenecen a bancos, tarjetas de créditos, servicios de compra/venta online, entre otros. Incluso también para los servicios web que no se publican, son privados y necesitan mayor control y seguridad como los que pertenecen a empresas privadas y redes militares.

Empresas competidoras pueden aprender el know-how y conseguir copiar el diseño para ofrecer servicios similares y competitivos. Pero no solo se trata de competencia, los ataques de seguridad como espionaje de información, suplantación de clientes, inyección de comandos y denegación de servicio también son posibles ya que los atacantes pueden aprender sobre los datos intercambiados y los patrones de invocación de los documentos WSDL. Si bien la legibilidad de las descripciones de los servicios hace que los servicios web sean reconocibles, también contribuye a la vulnerabilidad del servicio [2].

Suena lógico entonces analizar la seguridad que posee un determinado WSDL para controlar la seguridad del mismo. Para determinar esto, es necesario primero comprender el SW mediante el análisis de su correspondiente especificación WSDL.

Aprovechando la alta estandarización presente en el lenguaje WSDL, se pueden definir metodologías de comprensión de los mismos mediante la extracción y análisis de la información presente en dichas especificaciones [3, 4, 5]. A partir de esta información, determinar qué tan entendible es un WSDL y qué vulnerabilidades puede reflejar. Luego, usando diferentes técnicas, se pueden disminuir estas vulnerabilidades, ocultándolas o, en casos extremos, forzando la reconstrucción del WSDL o su SW asociado [6].

En este trabajo se extrae información aplicando técnicas de compilación, algoritmos de análisis de lenguaje natural y técnicas de cálculo de indicadores sobre su especificación WSDL. Toda esta información se utiliza para, a partir del cálculo de métricas propias, determinar la dificultad de comprensión que poseen [7, 8]. Además, también se utiliza LSP (Logic Scoring of Preference) para definir estructuras de agregación que le asignen pesos a los valores de cada métrica según sea la necesidad del ingeniero de software y, en base a estos pesos, se obtiene un grado de entendimiento global de la especificación WSDL [9,10]. Toda esta información se usa de base para definir qué partes del WSDL muestran vulnerabilidades y, qué modificaciones se pueden realizar sobre el mismo para mitigar esto.

La organización de este artículo se expone a continuación. La sección 2 describe la línea de investigación abordada. La sección 3 presenta los resultados obtenidos hasta el momento, junto con aquellos esperados a corto plazo. Finalmente, la sección 4 describe las tareas realizadas por los recursos humanos en formación.

Líneas de Investigación y Desarrollo

El análisis y reducción de vulnerabilidades de las especificaciones WSDL posee múltiples etapas con su función particular dentro del proceso global. En las subsecciones siguientes se describen brevemente dichas etapas.

Extracción de Información

Debido a que las especificaciones WSDL son un dialecto XML, se pueden usar técnicas de compilación sobre las mismas basadas en los parser DOM (Domain Object Model) [11]. Un parser DOM construye una representación interna del WSDL basada en árboles. A partir de funciones específicamente diseñadas para recorrer la representación construida (funciones

transversales) se extrae la información deseada. Estas funciones transversales logran extraer los identificadores de cada componente (*name, type element*, *etc.*), la documentación y los comentarios presentes en el WSDL.

Cálculo de Métricas
De la información extraída del WSDL, se calculan múltiples métricas, que pueden ser lógicamente diferenciadas en los siguientes grupos:

- **Métricas de tamaño:** miden la complejidad del WSDL en base a las cantidades de componentes de cada etiqueta dentro del WSDL, como por ejemplo cantidad de tipos complejos, de parámetros, de operaciones, de mensajes, entre otras. Esto permite tener una idea del tamaño de cada sección particular del WSDL y determinar qué tan complejo es, a primera vista, su comprensión.
- **Métricas de calidad:** permiten medir la calidad semántica de la especificación WSDL. Esto es, que tanta información semántica brinda la especificación WSDL respecto del WS que representa y que tan entendible y comprensible en sí es dicha especificación.
- Métrica de entendimiento global: usando LSP se calcula el grado de entendimiento que posee la especificación WSDL de un WS [9, 10].

Incrementar la Seguridad
Toda aplicación web está conformada por distintos tipos de información, tanto formal como informal. El análisis detallado de la misma y el cálculo de métricas permite detectar, dado su grado de entendimiento, que partes son más susceptibles a los ataques. En este punto, es posible definir estrategias que permitan subsanar las vulnerabilidades y proteger las partes que sean susceptibles de ataques [6].

Utilizando la información extraída del WSDL se pueden manipular diferentes partes del mismo para mejorar su seguridad disminuyendo su nivel de entendimiento. Esto se puede lograr mediante la utilización de funciones de ofuscación y/o encriptación al realizar las modificaciones y/o transformaciones necesarias que aumentaran el nivel de seguridad. Estas transformaciones pueden ser sobre partes específicas del WSDL (identificadores, operaciones, etc.) o en la totalidad del mismo. Dichas modificaciones dependen del nivel de seguridad deseado, partiendo de un nivel básico en donde se ofuscan y/o encriptan partes específicas del WSDL, como por ejemplo el nombre de los identificadores, hasta llegar a un nivel máximo en donde se realiza una transformación completa del WSDL.

Resultados Obtenidos/Esperados

Algunos de los resultados más destacados obtenidos por esta investigación son:

- Se definieron y calcularon diferentes métricas (cuantitativas y cualitativas) que permiten medir la complejidad de los WS.
- Se utilizó LSP para calcular el grado de entendimiento global del WS.
- Se construyó WSDLUD, una herramienta que automatiza el proceso de cálculo de métricas, del grado de entendimiento del WS usando LSP y la visualización de la información.
- Se definieron métodos y herramientas para la reducción de vulnerabilidades del WSDL mediante su ofuscación y/o encriptación.

Entre los objetivos planteados a corto y largo plazo se pueden mencionar:

- Mejorar las técnicas de mitigación de vulnerabilidades.
- Construir una herramienta que, vinculada con WSDLUD, permite realizar la transformaciones de ofuscación/encriptación de manera automática.
- Ampliar y aplicar el prototipo a especificaciones escritas en BPEL debido a que este lenguaje es utilizado para la ejecución de procesos de negocios.
- Estudiar, comprender y ampliar el número de métodos de encriptación y ofuscación de código utilizados.

Formación de Recursos Humanos

Las tareas realizadas en el contexto de la presente línea de investigación están siendo desarrolladas como parte de trabajos para optar al grado de Magister en Ingeniería de Software. En el futuro se piensa generar diferentes tesis de maestría y doctorado a partir de los resultados obtenidos de este trabajo.

Bibliografía

[1] WSDL Specification for W3C https://www.w3.org/TR/wsdl.

[2] Pananya Sripairojthikoon, Twittie Senivongse. “Concept-Based Readability Measurement and Adjustment for Web Services Descriptions”. ICACT Transactions on Advanced Communications Technology (TACT) Vol. 3, Issue 1, January 2014.

[3] N. Gold and K. Bennett. "Program com-prehension for web services". In Program Comprehension, 2004. Proc. 12th IEEE International Workshop on. June 2004.

[4] L. O’Brien Lero and D. Smith. "Working session: program comprehension strategies for web service and service oriented archi-tectures". Proc. of 12th IEEE International Workshop on Program Comprehension. 2004.

[5] H. El Bouhissi, M. Malki, and D. Bouchiha. "A reverse engineering approach for the web service modeling ontology specifications". In Sensor Technologies and Applications2008. SENSORCOMM ’08. Second International Conferenceon, pages 819–823, Aug 2008.

[6] Edgardo Bernardis, Hernán Bernardis, Mario Berón, Germán Montejano. “Seguridad en Servicios Web”. XIX Workshop de Informática y Ciencias de la Computación (WICC). Buenos Aires, Argentina. Abril de 2017.

[7] C. Mao. "Towards a data complexity metric set for web service composition". In Computer and Information Technology(CIT), 2011 IEEE 11th International Conference on, pages 127–131, Aug 2011.

[8] Fangfang Liu, Yuliang Shi, Jie Yu, Tianhong Wang, Jingzhe Wu. “Measuring Similarity of Web Services Based on WSDL”. IEEE International Conference on Web Services ICWS. 2010.

[9] Bernardis, Hernán; Berón Mario; Bernardis, Edgardo; Riesco, Daniel; Henriques, Pedro. “Extracción de información y cálculo de métricas en WSDL 1.1 y 2.0”. II Congreso Nacional de Ingeniería Informática / Sistemas de información (CoNaIISI). Argentina. 2014.

[10] Mario M. Berón, Hernán Bernardis, Enrique A. Miranda, Daniel E. Riesco, Maria João Pereira, Pedro Rangel Henriques. "WSDLUD: a Metric to Measure the Understanding Degree of WSDL Descriptions". Proceedings of the 2015 Symposium on Languages, Applications and Technologies, SLATE'15. Madrid, España 2015.

[11] Parser DOM specification for W3C. https://www.w3.org/DOM.